Spooky Author Identification

NLA project

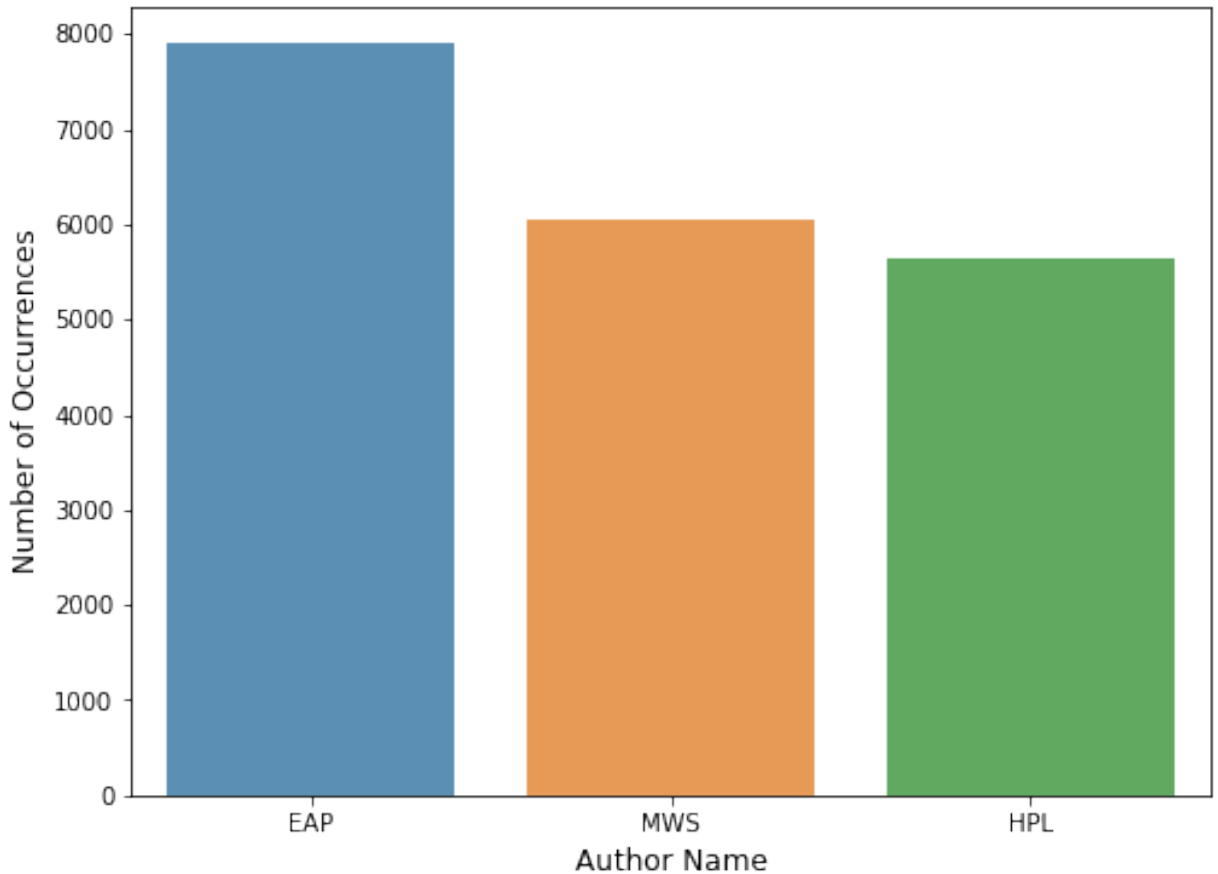
Bakers

Problem

- There are a lot of excerpts from horror stories written by Edgar Allan Poe, Mary Shelley, and HP Lovecraft.
- The chalenge was to predict the author of the cite using the excerpts with authorship.
- Use NLA methods to predict authors.

Train & Test Data

- Train sample consists of excerpt ids, excerpt itself and author. Use 80% of the dataset to train our methods.
- The other data is used for testing. For each excerpt we estimate the probabilities that the cite belongs to current author.


8000
7000
6000
5000
4000
3000
2000
1000
0
Number of Occurrences
EAP
MWS
HPL
Author Name

Accuracy

- Compare the predicted aithors with real ones and calculate the mean.
- Estimate log-loss for each method:

$$logloss = -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{M} y_{i,j}\,\ln(p_{i,j})$$

The First Approach

Direct SVD application 1

1. Obtain and consider the matrices (terms × docs) and (terms × authors).
2. Find basis of authors × terms matrix using SVD.
3. Find a projection of each document from (terms × docs) matrix to 3-D subspace V.

$$doc_p = \Sigma_r^{-1} U_r^\top doc$$

4. Estimate **cosine** between projected target document and each author.

$$\text{cosine} = \frac{(doc_p, author_j)}{\|doc_p\| \|author_j\|}$$

5. Transform cosine to probabilities.

$$P(author_j = i) = \frac{\text{cosine}\ (q_j, a_i)}{\sum_{k=1}^{3} \text{cosine}\ (q_j, a_k)}$$

Accuracy

1. Logloss: 1.103822539347749
2. Similarity between predicted and true labels: 52.35%

Direct SVD application 2

1. Obtain and consider the matrices (terms × docs) and (terms × authors).

2. Find basis of traim terms × docs matrix using SVD.

3. Find a projection of each test document from (terms × docs) matrix to subspace V.

$$doc_p = \Sigma_r^{-1} U_r^\top doc$$

4. Find a projection of each document from (terms × authors) matrix to subspace V.

$$author_p = \Sigma_r^{-1} U_r^\top author$$

5. Estimate **cosine** between projected target document and each author.

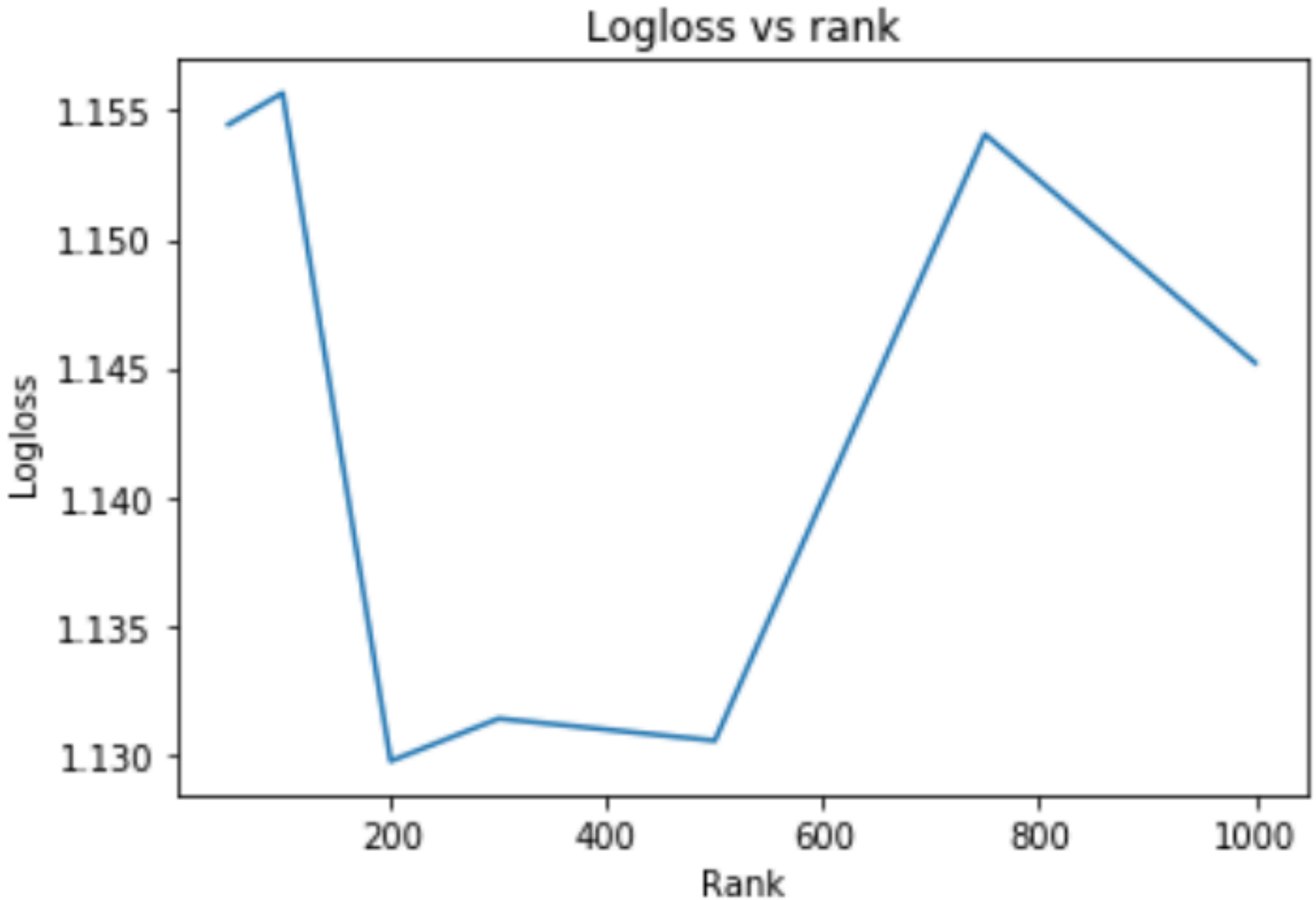
$$\text{cosine} = \frac{(doc_p, author_j)}{\|doc_p\| \|author_j\|}$$

6. Transform cosine to probabilities.

$$P(author_j = i) = \frac{\text{cosine}\,(q_j, a_i)}{\sum_{k=1}^{3} \text{cosine}\,(q_j, a_k)}$$

Accuracy

The best Logloss: 1.12975999278


Logloss vs rank
1.155
1.150
1.145
1.140
1.135
1.130
Logloss
200
400
600
800
1000
Rank

The Second Approach

In this approach we consider a system of equations Ax = b, where A is a matrix of the size num_of_texts × num_of_different_words and contains the frequency of words in different texts. b is a target vector of authors of the size num_of_texts × 1.

The system can be solved using truncated SVD decomposition of A: A = USV^T.
$A^{-1} = VS^{-1}U^T$.

Hence, the vector of parameters will be x = $VS^{-1}U^Tb$.

Then, we can obtain matrix A_{test} similar to the matrix A, but for the test-data. And, finally, calculate the taget vector b_{test} = A_{test} x.

Accuracy

1. Logloss: 0.8750022844199319
2. Similarity between predicted and true labels: 58.35%

Kinetic features

We've also tried to use "kinetic" features of the text. Let's intoduce a notion of the kinetic feature. For instance, we have a text "a a a e u". Then, it is said that the word 'a' has the probability of 3/5, 'e' and 'u' 1/5. And the kinetic of the text is the sum of squared probabilities. Thus, if all words are the same, the kinetic will be 1. If all words are different, then the kinetic will be 1/N, where N is the number of words.

Hence, the kinetic feature of the text may show how different or similar words are. However, it's not the only possible way to use this approach. We can calculate the kinetic of letters, punctuation signs, vowel and conconant sounds, and kinetic of parts of the speech. The last may show, for example, that an author uses much more adjectives and adverbs than others to describe something. So, his kinetic feature of parts of the speech will be closer to 1 (because there are more similar words of the same part of the speech).

Reference:

"Experimented Kinetic Energy as Feature for Natural Language Classification" - Daia Alexandru, Quentin Garnier, Wanjiku Francis, Solomon Mbandi.

Accuracy

Log loss: 1.2210514846494618

The Third Approach

Finding of the SVD basis

Let $A \in \mathbb{C}^{n \times m}$ be of rank r where n is a number of texts of certain author and m number of words in all texts. Matrix A consist of TF-IDF representations of texts.

Let $A = U\Sigma V^*$ be its SVD and we know that $\text{im}(A^*) = \text{span}\{v_1, \dots, v_r\}$, where $V = [v_1, \dots, v_n]$.

For each author we try to find basis in subspace V_k with dimensionality k. Then we try to decompose test vector b in basis of each author $\{v_{i1}, \dots, v_{ik}\}$ and find residual of the decomposition.

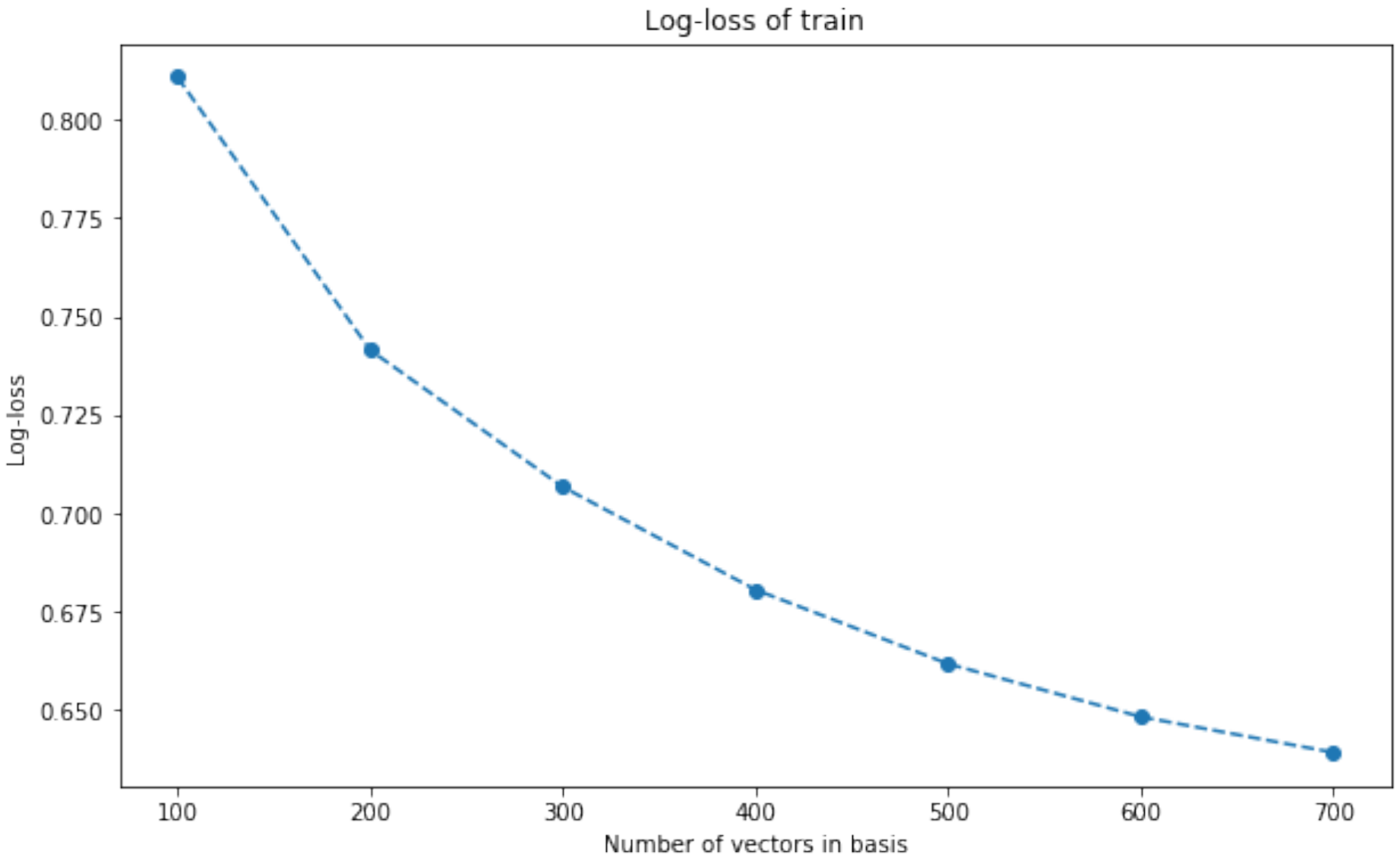
For $i = 1, 2, 3$.

$$r_i = \min_C ||b - Cv_i||_2$$

The we can say that less residual then less distance from this author to author of the test text.

The we can assign probabilites og belonging to each class. We will do it with this function:

$$\mathbb{P}(author = i) = \frac{\exp(-\alpha r_i)}{\sum_j \exp(-\alpha r_o)}$$


Log-loss of train
0.800
0.775
0.750
0.725
0.700
0.675
0.650
Log-loss
100
200
300
400
500
600
700
Number of vectors in basis

The Fourth Approach

Finding Krylov basis

Let $A \in \mathbb{C}^{n \times m}$ be of rank r where n is a number of texts of certain author and m number of words in all texts. Matrix A consist of TF-IDF representations of texts.

For each author we will try to find subspace with dimensionality k with Krylov basis $\hat{V_k}$.

For $i = 1, 2, 3$.

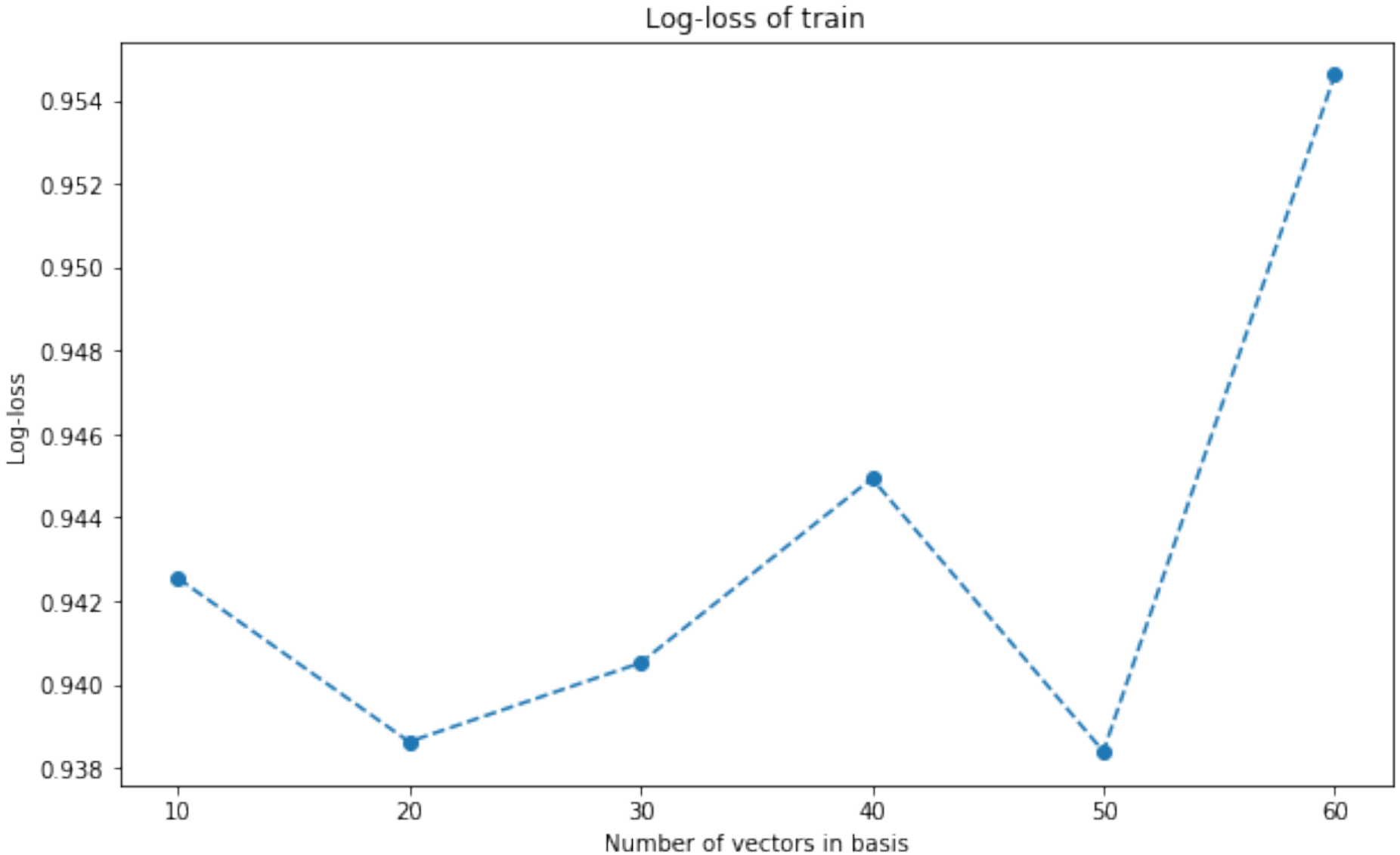
$$\begin{aligned} \hat{v_{i0}} &= x_0 \\ \hat{v_{i1}} &= A_i x_0 \\ &\dots \\ \hat{v_{ik}} &= A_i^k x_0 \end{aligned}$$

Then we ortogonalize this basis with Gram Schmidt process and obtain basis $\{v_{i1}, \dots, v_{ik}\}$.

Now we try to decompose test vector b in basis of each author $\{v_{i1}, \dots, v_{ik}\}$ and find residual of the decomposition.

$$r_i = \min_C ||b - Cv_i||_2$$

The we can say that less residual then less distance from this author to author of the test text.


Log-loss of train
0.954
0.952
0.950
0.948
0.946
0.944
0.942
0.940
0.938
Log-loss
10
20
30
40
50
60
Number of vectors in basis

Conclusion

We've tried three different approaches and got the following results:

1.
   - SVD_1_1 *Log loss: 1.104*
   - SVD_1_2 *Log loss: 1.1298*
2. Linear System, Kinetic Features *Log loss: 0.8750*
3. SVD_2: *Log loss: 0.633*
4. Krylov subspace: *Log loss: 0.939*

Thus, the most appropriate way we found for such problems is to use SVD the following way: find the best approximating subspace for each author, decompose test vector in it and then compare the residuals.